A Theoretically-Principled Sparse, Connected, and Rigid Graph Representation of Molecules

Shih-Hsin Wang
University of Utah

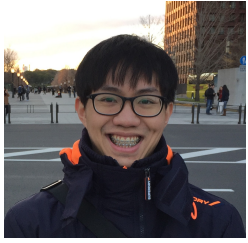
shwang@math.utah.edu

ICLR 2025 Oral Presentation

Shih-Hsin Wang*[1]

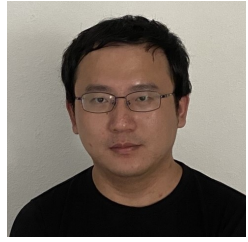
Yuhao Huang*[1]

Justin Baker[2]



Yuan-En Sun[3]

Qi Tang[4]

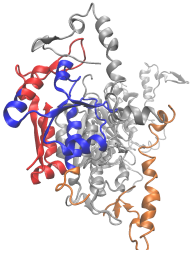


Bao Wang[1]

[1]Department of Mathematics and Scientific Computing and Imaging (SCI) Institute, University of Utah
[2]Department of Mathematics, UCLA
[3]Department of Biochemistry, University of Utah
[4]School of Computational Science and Engineering, Georgia Tech

Motivation: Why Graph Structure Matters

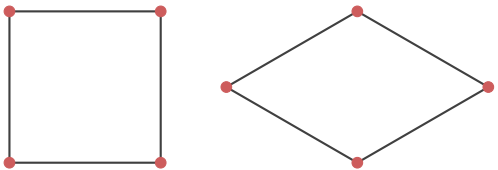
- Molecules as 3D point clouds
- Need graph structure to apply GNNs
- Geometric graph's properties $\rightarrow$ GNN's performance



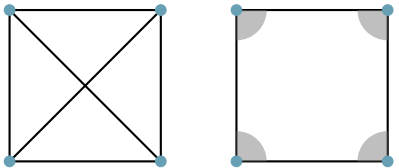
What Makes a Good Geometric Graph?

1 Introduction

- **Sparsity**: # of edges $\downarrow \Rightarrow$ computational efficiency $\uparrow$
- **Connectedness**: ensures message passing across all nodes
- **Rigidity**: uniquely determine 3D geometry



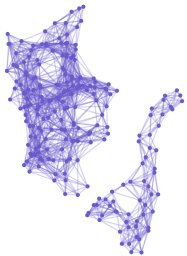
Non-rigid graphs



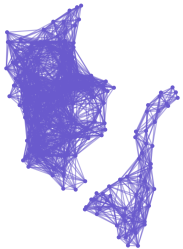
Rigid graphs

Limitations of Existing Graph Constructions

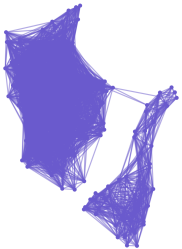
- **No method satisfies sparsity, connectedness, and rigidity simultaneously.**
- Take **radial cutoff graph** as an example:
  - *Small cutoff* $\rightarrow$ sparse but disconnected
  - *Large cutoff* $\rightarrow$ connected but dense



Cutoff $= 8$



Cutoff $= 12$



Cutoff $= 16$

Our Goals

1 Introduction

- Build graphs that are: **Sparse**, **Connected**, **Rigid**
- **No hyperparameter** tuning
- Theoretical guarantees for GNN performance

SCHull (Ours)

Our Solution: Spherical Convex Hull (SCHull)

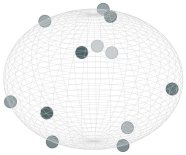
2 Methods

- **Step 1**: Center the point cloud and project each point onto the unit sphere.

Our Solution: Spherical Convex Hull (SCHull)

2 Methods

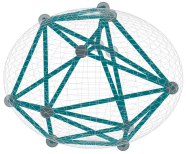
- **Step 2**: Build the convex hull of projected points.



Our Solution: Spherical Convex Hull (SCHull)

2 Methods

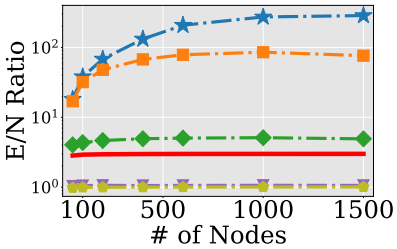
- **Step 3**: Convex hull's edges define the graph's connections.



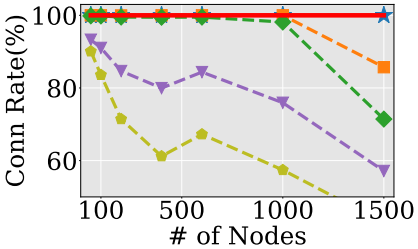
Why It Works

2 Methods

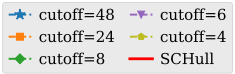
- **Hyperparameter-Free**: No tuning required.
- **Sparsity & Connectedness**: Convex hull gives a connected graph with $\#\text{Edges} < 3 \cdot \#\text{Nodes}$.



Edge-to-node ratio



Percent of connected graphs



Why It Works

2 Methods

- **Rigidity**: Geometry is restored via:
  - Edges: edge length & dihedral angle (from convex hull)
  - Nodes: distance to center

- **Theoretical Guarantee**: with mild assumptions,

  1-*layer GNN can distinguish SCHull graphs of any two point clouds up to rigid motion.*

	1 Layer	2 Layers	# Edges / # Nodes	# Components
Radius Graph ($r = 1.8$) w/ d_{ij}	50.0 ± 0.0	50.0 ± 0.0	1.0	(1, 2)
Radius Graph ($r = 2.5$) w/ d_{ij}	50.0 ± 0.0	51.0 ± 3.0	3.0	1
Radius Graph ($r = 3.0$) w/ d_{ij}	60.5 ± 11.1	57.2 ± 9.1	5.0	1
Complete Graph w/ d_{ij}	$59.0 \pm 6.6.0$	50.0 ± 0.0	6.5	1
SCHull w/ d_{ij}	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	2.57	1
SCHull w/ d_{ij} and τ_{ij}	$\mathbf{100.0 \pm 0.0}$	$\mathbf{100.0 \pm 0.0}$	2.57	1

Table: Comparison of GNN performance (Unit:%) on distinguishing 14-point symmetric point clouds.

Numerical Results: MD17

3 Experiments

Molecule	DimeNet	DimeNet+SCHull	SphereNet	SphereNet+SCHull	LEFTNet	LEFTNet+SCHull
Aspirin	0.499	$0.427_{\pm.004}$	0.430	$0.387_{\pm.005}$	0.281	$\mathbf{0.240}_{\pm.005}$
Benzene	0.187	$0.157_{\pm.006}$	0.178	$0.155_{\pm.004}$	0.147	$\mathbf{0.098}_{\pm.002}$
Ethanol	0.230	$0.198_{\pm.003}$	0.208	$0.181_{\pm.003}$	0.138	$\mathbf{0.109}_{\pm.002}$
Malonaldehyde	0.383	$0.334_{\pm.003}$	0.340	$0.298_{\pm.003}$	0.205	$\mathbf{0.151}_{\pm.002}$
Naphthalene	0.215	$0.178_{\pm.002}$	0.178	$0.144_{\pm.002}$	0.074	$\mathbf{0.058}_{\pm.001}$
Toluene	0.210	$0.169_{\pm.002}$	0.155	$0.129_{\pm.002}$	0.083	$\mathbf{0.076}_{\pm.001}$
Uracil	0.301	$0.288_{\pm.002}$	0.267	$0.242_{\pm.003}$	0.117	$\mathbf{0.095}_{\pm.001}$
Training Time/Epoch(s)	$43_{\pm 0.9}$	$50_{\pm 0.8}$	$51_{\pm 1.0}$	$62_{\pm 1.5}$	$24_{\pm 0.5}$	$28_{\pm 0.5}$

Table: Test MAEs of MD17 dataset vector-valued properties prediction.

Numerical Results: Fold & React

3 Experiments

Method	React	Avg. Time	Fold				Avg. Time
			Fold	Super	Family	Avg.	
GCN [5]	67.3		16.8	21.3	82.8	40.3	–
IEConv [3]	87.2	–	45.0	69.7	98.9	71.2	–
DWNN [6]	76.7	–	31.8	37.8	85.2	51.5	–
GearNet [9]	79.4	–	28.4	42.6	95.3	55.4	–
HoloProt [7]	78.9	–	–	–	–	–	–
MACE [1]	–	–	$23.7_{\pm 0.5}$	$21.4_{\pm 0.5}$	$60.2_{\pm 0.2}$	35.1	$114_{\pm 0.5}$
MACE+SCHull	–	–	$27.0_{\pm 0.6}$	$23.1_{\pm 0.5}$	$65.0_{\pm 0.2}$	38.4	$135_{\pm 0.5}$
SEGNN [2]	–	–	$28.8_{\pm 0.6}$	$30.3_{\pm 0.6}$	$77.1_{\pm 0.3}$	45.4	$121_{\pm 0.7}$
SEGNN+SCHull	–	–	$32.0_{\pm 0.4}$	$36.8_{\pm 0.7}$	$86.9_{\pm 0.3}$	51.9	$152_{\pm 0.5}$
GVP-GNN [4]	65.5	$320_{\pm 5}$	16.0	22.5	83.8	40.8	$106.3_{\pm 0.5}$
GVP-GNN + SCHull	$77.1_{\pm 0.5}$	$345_{\pm 5}$	$24.5_{\pm 0.3}$	$27.1_{\pm 0.2}$	$88.6_{\pm 0.3}$	46.7	$111.5_{\pm 0.5}$
ProNet-Amino-Acid [8]	86.0	$210_{\pm 5}$	51.5	69.9	99.0	73.5	$70.5_{\pm 0.5}$
ProNet-Amino Acid+SCHull	$87.9_{\pm 0.3}$	$221_{\pm 6}$	$55.2_{\pm 0.2}$	$73.9_{\pm 0.2}$	$99.1_{\pm 0.1}$	76.1	$73.8_{\pm 0.5}$
ProNet-Backbone [8]	86.4	$213_{\pm 5}$	52.7	70.3	99.3	74.1	$71.4_{\pm 0.8}$
ProNet-Backbone+SCHull	$\mathbf{88.1}_{\pm 0.3}$	$230_{\pm 5}$	$\mathbf{56.1}_{\pm 0.3}$	$\mathbf{74.6}_{\pm 0.2}$	$\mathbf{99.4}_{\pm 0.1}$	**76.7**	$75.8_{\pm 0.5}$

Table: Accuracy (%) on protein fold and enzyme reaction classification tasks. Ave. Time denotes the average time per training epoch. The top results are in boldface. SCHull consistently improves baseline models.

Numerical Results: LBA

3 Experiments

Method	LBA				Avg. Time
	RMSE↓	Pearson↑	Spearman↑	Kendall↑	
IEConv [3]	1.554	0.414	0.428	–	–
HoloProt-Full Surface [7]	1.464	0.509	0.500	–	–
HoloProt-Superpixel [7]	1.491	0.491	0.482	–	–
GVP-GNN [4]	$1.529_{\pm 0.001}$	$0.441_{\pm 0.001}$	$0.432_{\pm 0.002}$	$0.301_{\pm 0.002}$	$48.6_{\pm 0.6}$
GVP-GNN + SCHull	$1.401_{\pm 0.001}$	$0.475_{\pm 0.001}$	$0.459_{\pm 0.001}$	$0.335_{\pm 0.002}$	$53.6_{\pm 0.6}$
ProNet-Amino-Acid [8]	1.455	0.536	0.526	$0.465_{\pm 0.001}$	$31.7_{\pm 0.5}$
ProNet-Amino Acid+SCHull	$1.355_{\pm 0.002}$	$0.556_{\pm 0.001}$	$0.568_{\pm 0.001}$	$0.512_{\pm 0.001}$	$33.9_{\pm 0.5}$
ProNet-Backbone [8]	1.458	0.546	0.550	$0.481_{\pm 0.001}$	$32.1_{\pm 0.5}$
ProNet-Backbone+SCHull	$\mathbf{1.321_{\pm 0.002}}$	$\mathbf{0.581_{\pm 0.001}}$	$\mathbf{0.578_{\pm 0.1}}$	$\mathbf{0.535_{\pm 0.001}}$	$34.4_{\pm 0.5}$

Table: RMSE/Pearson Correlation/Spearman Correlation/Kendall Correlation on the LBA Test Dataset. Ave. Time refers to the average running time of one epoch in model training.

When to Use SCHull

4 Discussion

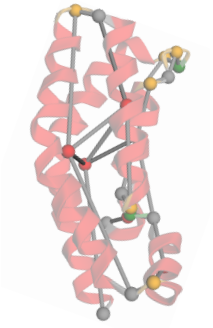
- **Struggling to balance sparsity, connectedness, and rigidity?**
  SCHull provides a principled solution — no hyperparameters needed.
- **How to apply SCHull:**
  - **Replace:** Use SCHull alone.
  - **Augment:** Combine with domain-specific (sparse) graphs.
- **Beyond Molecules:** Plug SCHull into any 3D graph learning pipeline — minimal cost, broad utility.
- **Takeaway:**

A lightweight, geometry-aware graph — usable with or without existing graphs.

Use Cases: Where SCHull Helps

4 Discussion

- **Fragment-based Pipelines**
  - *Issue:* Fragmentation causes variable distances between nodes.
  - *SCHull:* Builds edges without global thresholds.
- **LLM-based Models**
  - *Issue:* LLMs lack inductive bias for 3D geometry.
  - *SCHull:* Provides lightweight, geometry-aware graphs.



Fragmentation Graph

Fragments as nodes

Empirical Evidence: React & LBA

4 Discussion

Method	Acc.	Ave.Time
GVP-GNN	65.5	320
ProNet-Backbone	86.4	213
Fragment + SCHull	87.2	**116**
Fragment + SCHull + Mamba	**88.4**	157

Table: Accuracy (%) on enzyme reaction classification tasks. Ave. Time denotes the average time per training epoch.

Method	RMSE (↓)	Pearson (↑)	Spearman (↑)	Ave.Time (↓)
GVP-GNN	1.529	0.441	0.432	49
ProNet-Backbone	1.458	0.546	0.550	32
Fragment + SCHull	1.435	0.579	0.591	**24**
Fragment + SCHull + Mamba	**1.399**	**0.614**	**0.610**	29

Table: RMSE/Pearson Correlation/Spearman Correlation on the LBA Test Dataset.

Summary & Code Access

4 Discussion

- **SCHull Graph Construction**:
  - Sparse, connected, and rigid.
  - Hyperparameter-free and theoretically principled.
  - Consistently strong across both synthetic and real-world tasks.
- **Resources**:
  - **Code, paper, and slides** available online.
  - Scan the QR code or visit my website to try it out.

References

5 References

Ilyes Batatia, Dávid Péter Kovács, Gregor NC Simm, Christoph Ortner, and Gábor Csányi.
Mace: Higher order equivariant message passing neural networks for fast and accurate force fields.
arXiv preprint arXiv:2206.07697, 2022.

Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J Bekkers, and Max Welling.
Geometric and physical quantities improve e (3) equivariant message passing, 2021.
URL https://arxiv. org/abs/2110.02905.

Pedro Hermosilla, Marco Schäfer, Matěj Lang, Gloria Fackelmann, Pere Pau Vázquez, Barbora Kozlíková, Michael Krone, Tobias Ritschel, and Timo Ropinski.
Intrinsic-extrinsic convolution and pooling for learning on 3d protein structures.
arXiv preprint arXiv:2007.06252, 2020.

Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael John Lamarre Townshend, and Ron Dror.
Learning from protein structure with geometric vector perceptrons.
In *International Conference on Learning Representations*, 2020.

References

5 References

Thomas N. Kipf and Max Welling.
Semi-supervised classification with graph convolutional networks.
In *Proceedings of the 5th International Conference on Learning Representations*, 2017.

Jiahan Li.
Directed weight neural networks for protein structure representation learning.
arXiv preprint arXiv:2201.13299, 2022.

Vignesh Ram Somnath, Charlotte Bunne, and Andreas Krause.
Multi-scale representation learning on proteins.
Advances in Neural Information Processing Systems, 34:25244–25255, 2021.

Limei Wang, Haoran Liu, Yi Liu, Jerry Kurtin, and Shuiwang Ji.
Learning hierarchical protein representations via complete 3d graph networks.
arXiv preprint arXiv:2207.12600, 2022.

References

5 References

Zuobai Zhang, Minghao Xu, Arian Jamasb, Vijil Chenthamarakshan, Aurelie Lozano, Payel Das, and Jian Tang.
Protein representation learning by geometric structure pretraining.
arXiv preprint arXiv:2203.06125, 2022.